

This listing of claims will replace all prior versions, and listings of claims in the application:

**Listing of Claims:**

1. (currently amended) A platform-independent audio computer service capable of servicing platform dependent audio events ~~[[evens]]~~ on a first and a second platform, the platform-independent audio computer service comprising:

an audio component capable of receiving as input an audio event, the audio event representing an event which requires an audio response, the audio component running on a first platform serviced by the ~~Platform~~ platform independent audio computer service; and

a software object representing a set of entries, wherein at least one entry of the set of entries is associated with the audio event and a first theme representing an audio cue that maps to the audio event, the first theme including a first set of platform dependent audio fields, each platform dependent audio field associated with at least one platform dependent audio event of the first platform, wherein the first theme is arranged to permit the emulation of the audio event of the first platform in response to the audio event; and

an audio system manager capable of directly or indirectly accessing the software object and audio component, thereby allowing the platform dependent audio event to be emulated using the software object.

2. (previously presented) The audio computer service of claim 1 wherein the audio computer service is implemented in Java programming language.

3. (original) The audio computer service of claim 1 wherein the audio event is a GUI audio event.

4. (original) The audio computer service of claim 1 wherein the first platform is an operating system.

5. (currently amended) The audio computer service of claim 4 wherein the operating system is one of a Windows operating system, a Motif operating system or a ~~MacIntosh~~ Macintosh operating system.

6. (original) The audio computer service of claim 1 wherein the software object is stored in volatile memory.

7. (original) The audio computer service of claim 1 further including a second platform serviced by the computer service.

8. (original) The audio computer service of claim 7 further including a second theme including a second set of platform dependent audio fields.

9. (original) The audio computer service of claim 7 wherein the first and second theme are arranged to permit the emulation of audio events of different graphical user interfaces.

10. (original) The audio computer service of claim 1 further including a Multiplexer.

11. (original) The audio computer service of claim 1 wherein one platform dependent audio field in the first set of platform dependent audio fields includes a pointer.

12. (currently amended) A computer-implemented method of accessing, by a platform-independent audio computer service, a platform dependent audio field associated with an audio event from a first platform, the method comprising:

receiving a request for a platform dependent audio function;

importing, using the platform-independent audio computer service, a theme corresponding to the platform dependent audio function including at least one platform dependent audio field associated with the platform dependent audio function when the request is received; and

referencing the platform dependent audio field corresponding to the platform dependent audio function based on the imported theme.

13. (original) The method of 12 further including accessing a platform dependent audio file corresponding to the platform dependent audio field.

14. (previously presented) The method of 12 wherein the importing of the theme corresponding to the platform dependent audio field uses a Multiplexer.

15. (original) The method of 12 further including adding a listener to a component which provides the audio event.

16. (previously presented) A software object for servicing audio events by a platform-independent audio computer service, the object comprising:

a first set of platform dependent fields which can provide audio output for a first platform; and

a set of audio events, each audio event associated with at least one platform dependent field of the first set of platform dependent fields, wherein the first set of platform dependent fields are included in a first theme which relates the first set of platform dependent fields to a first platform.

17. (previously presented) The software object of claim 16 further including a second theme which includes a second set of platform dependent fields for a second.

18. (currently amended) The software object of claim 16 wherein the first platform is one of a Windows look and feel, a Motif look and feel or a ~~MacIntosh~~ Macintosh look and feel.

19. (original) The software object of claim 16 wherein the software object is a hash file.

20. (original) The software object of claim 16 wherein the set of audio events is organized into categories.

21. (original) The software object of claim 16 wherein the software object is a hash file.

22. (previously presented) A platform-independent audio computer service comprising:

a system manager;

a component capable of an audio event designed to run on a first platform serviced by the audio computer service; and

a software object having a set of entries, wherein at least one entry is associated with the audio event, a first theme and a second theme, the first theme including a first set of platform dependent audio fields, each platform dependent audio field of the first theme associated with at least one audio event, the second theme including a second set of platform dependent audio fields, each platform dependent audio field of the second theme associated with at least one audio event wherein the first and second themes are arranged to permit the emulation of audio events of different graphical user interfaces.

23. (Canceled)

24. (Canceled)

25. (New) A method of servicing platform dependent audio events, by a platform-independent audio computer service, a platform dependent audio field associated with an audio event on a first and a second platform, the method comprising:

providing an audio component capable of receiving as input an audio event, the audio event representing an event which requires an audio response, the audio component running on a first platform serviced by the platform independent audio computer service; and

providing a software object representing a set of entries, wherein at least one entry of the set of entries is associated with the audio event and a first theme representing an audio cue that maps to the audio event, the first theme including a first set of platform dependent audio fields, each platform dependent audio field associated with at least one platform dependent audio event of the first platform, wherein the first theme is arranged to permit the emulation of the audio event of the first platform in response to the audio event; and

providing an audio system manager capable of directly or indirectly accessing the software object and audio component, thereby allowing the platform dependent audio event to be

emulated using the software object.

26. (New) A computer-implemented method of accessing, by a platform-independent audio computer service, a platform dependent audio field associated with an audio event from a first platform, the method comprising:

providing a software object representing a set of entries, wherein at least one entry of the set of entries is associated with the audio event and a theme representing an audio cue that maps to the audio event, the theme including a first set of platform dependent audio fields, each platform dependent audio field associated with at least one platform dependent audio event of the first platform, wherein the theme is arranged to permit the emulation of the audio event of the first platform in response to the audio event;

receiving a request for a platform dependent audio function;

importing, using the platform-independent audio computer service, a theme corresponding to the platform dependent audio function including at least one platform dependent audio field associated with the platform dependent audio function when the request is received; and

referencing the platform dependent audio field corresponding to the platform dependent audio function based on the imported theme.